

AMENDMENTS TO THE SPECIFICATION

Please cancel original paragraphs [0049] and [0057] and substitute the following amended paragraphs:

[0049] Web controller 204 may be implemented as a servlet. The web controller 204 is responsible for security authorization and for dispatching user HTTP requests, which have been received from the browser 110A, to appropriate action classes. Actions 208 may be implemented as classes in an object-oriented environment such as Java®. The action classes 208 interact with service objects 102 ~~through layer-through~~ SOM interfaces that supply data from the service objects to the action classes in the form of service object parameters. FIG. 13 is a class diagram for an example action class. Based on such service object data, one or more widgets 210 are instantiated and are arranged into a specified layout within panels 212. When an action 208 completes processing, the web controller 204 forwards the user request to the server page generator 206, which renders the final user interface for display by browser 110A.

[0057] While certain embodiments support the Picasso user interface style, other embodiments can support any other style. Widgets 210 are used to generate the content area of a Picasso user interface; no tag libraries are used. Further, with a system that includes web page generator 202, developers do not need to code, create or modify JSP pages, servlets, or actions, and need not know any web programming. Instead, developers can concentrate on developing business logic as embodied in service objects 102. The web page generator 202 interacts with program business logic through the service object interfaces, and dynamically generates the entire user interface including validation logic and data type conversions. As an example, typically several hundred of JSP pages, Action classes, and Bean classes are used in a network management application that is built on Picasso, and all such classes are created by the application developer. In the approach herein, only about ten JSP pages, Action classes, and

Bean classes are used. As a result, development ~~speed~~ delays and software maintenance costs are greatly reduced.